Robust Quantum Algorithms with ε-Biased Oracles

Tomoya Suzuki Shigeru Yamashita Masaki Nakanishi
Katsumasa Watanabe
Graduate School of Information Science, Nara Institute of Science and Technology
{tomoya-s, ger, m-naka, watanabe}@is.naist.jp

Abstract

This paper considers the quantum query complexity of *ε-biased oracles* that return the correct value with probability only $1/2+\varepsilon$. In particular, we show a quantum algorithm to compute N-bit OR functions with $O(\sqrt{N}/\varepsilon)$ queries to ε-biased oracles. This improves the known upper bound of $O(\sqrt{N}/\varepsilon^2)$ and matches the known lower bound; we answer the conjecture raised by the paper [13] affirmatively. We also show a quantum algorithm to cope with the situation in which we have no knowledge about the value of ε. This contrasts with the corresponding classical situation, where it is almost hopeless to achieve more than a constant success probability without knowing the value of ε.

1 Introduction

Quantum computation has attracted much attention since Shor's celebrated quantum algorithm for factoring large integers [15] and Grover's quantum search algorithm [11]. One of the central issues in this research field has been the *quantum query complexity*, where we are interested in both upper and lower bounds of a necessary number of oracle calls to solve certain problems [1, 3, 4, 5, 6, 8, 14]. In these studies, oracles are assumed to be *perfect*, i.e., they return the correct value with certainty.

In the classical case, there have been many studies (e.g., [10]) that discuss the case of when oracles are *imperfect* (or often called *noisy*), i.e., they may return incorrect answers. In the quantum setting, Høyer et al. [12] proposed an excellent quantum algorithm, which we call the *robust quantum search algorithm* hereafter, to compute the OR function of N values, each of which can be accessed through a quantum "imperfect" oracle. Their quantum "imperfect" oracle can be described as follows: When the content of the query register is x ($1 \leq x \leq N$), the oracle returns a quantum pure state from which we can measure the correct value of $f(x)$ with a constant probability. This noise model naturally fits into quantum subroutines with errors. (Note that most existing quantum algorithms have some errors.) More precisely, their algorithm robustly computes N-bit OR functions with $O(\sqrt{N})$ queries to an imperfect oracle, which is only a constant factor worse than the perfect oracle case. Thus, they claim that their algorithm does not need a serious overhead to cope with the imperfectness of the oracles. Their method has been extended to a robust quantum algorithm to output all the N bits by using $O(N)$ queries [9] by Buhrman et al. This obviously implies that $O(N)$ queries are enough to compute the parity of the N bits, which contrasts with the classical $\Omega(N \log N)$ lower bound given in [10].

It should be noted that, in the classical setting, we do not need an overhead to compute OR functions with imperfect oracles either, i.e., $O(N)$ queries are enough to compute N-bit OR functions even if an oracle is imperfect [10]. Nevertheless, the robust quantum search algorithm by Høyer et al. [12] implies that we can still enjoy the quadratic speed-up of the quantum search when computing

OR functions, even in the imperfect oracle case, i.e., $O(\sqrt{N})$ vs. $O(N)$. However, this is not true when we consider the probability of getting the correct value from the imperfect oracles *explicitly* by using the following model: When the query register is x, the oracle returns a quantum pure state from which we can measure the correct value of $f(x)$ with probability $1/2+\varepsilon_x$, where we assume $\varepsilon \leq \varepsilon_x$ for any x and we know the value of ε. In this paper, we call this imperfect quantum oracle *an ε-biased oracle* (or a biased oracle for short) by following the paper [13]. Then, the precise query complexity of the above robust quantum search algorithm to compute OR functions with an ε-biased oracle can be rewritten as $O(\sqrt{N}/\varepsilon^2)$, which can also be found in [9]. For the same problem, we need $O(N/\varepsilon^2)$ queries in the classical setting since $O(1/\varepsilon^2)$ instances of majority voting of the output of an ε-biased oracle is enough to boost the success probability to some constant value. This means that the above robust quantum search algorithm does not achieve the quadratic speed-up anymore if we consider the error probability explicitly.

Adcock et al. [2] first considered the error probability explicitly in the quantum oracles, then Iwama et al. [13] continued to study ε-biased oracles: they show the lower bound of computing OR is $\Omega(\sqrt{N}/\varepsilon)$ and the matching upper bound when ε_x are the same for all x. Unfortunately, this restriction to oracles obviously cannot be applied in general. Therefore, for the general biased oracles, there have been a gap between the lower and upper bounds although the paper [13] conjectures that they should match at $\Theta(\sqrt{N}/\varepsilon)$.

Our Contribution. In this paper, we show that the robust quantum search can be done with $O(\sqrt{N}/\varepsilon)$ queries. Thus, we answer the conjecture raised by the paper [13] affirmatively, meaning that we can still enjoy the quantum quadratic speed-up to compute OR functions even when we consider the error probability explicitly. The overhead factor of $1/\varepsilon^2$ in the complexity of the original robust quantum search (i.e., $O(\sqrt{N}/\varepsilon^2)$) essentially comes from the classical majority voting in their recursive algorithm. Thus, our basic strategy is to utilize *quantum amplitude amplification and estimation* [7] instead of majority voting to boost the success probability to some constant value. This overall strategy is an extension of the idea in the paper [13], but we carefully perform the quantum amplitude amplification and estimation in quantum parallelism with appropriate accuracy to avoid the above-mentioned restriction to oracles assumed in [13].

In most existing (classical and quantum) algorithms with imperfect oracles, it is implicitly assumed that we know the value of ε. Otherwise, it seems impossible to know when we can stop the trial of majority voting with a guarantee of a more than constant success probability of the whole algorithm. However, we show that, in the quantum setting, we can construct a robust algorithm even when ε is unknown. More precisely, we can estimate unknown ε with appropriate accuracy, which then can be used to construct robust quantum algorithms. Our estimation algorithm also utilizes quantum amplitude estimation, thus it can be considered as an interesting application of quantum amplitude amplification, which seems to be impossible in the classical setting.

2 Preliminaries

In this section, we introduce the quantum computing and the query complexity. We also define quantum biased oracles.

2.1 Quantum State and Evolution

A state of *n-qubit* quantum register $|\psi\rangle$ is a superposition of 2^n classical strings with length n, i.e., $|\psi\rangle = \sum_x \alpha_x |x\rangle$ where $x \in \{0,1\}^n$ and the *amplitudes* α_x are complex numbers consistent with the

normalization condition: $\sum_x |\alpha_x|^2 = 1$. If we *measure* the state $|\psi\rangle$ with respect to the standard basis, we observe $|x\rangle$ with probability $|\alpha_x|^2$ and after the measurement the state $|\psi\rangle$ collapses into $|x\rangle$.

Without measurements, a quantum system can evolve satisfying the normalization condition. These evolutions are represented by unitary transformations. The following *Fourier transform* is a famous example that acts on several qubits.

Definition 1 *For any integer $M \geq 1$, a quantum Fourier transform $\mathbf{F}_M$ is defined by*

$$\mathbf{F}_M \ : \ |x\rangle \ \longmapsto \ \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} e^{2\pi \imath xy/M} |y\rangle \quad (0 \leq x < M).$$

In this paper, unitary transformations controlled by other registers are often used. For example, one of them acts as some unitary transformation if the control qubit is $|1\rangle$, otherwise it acts as identity. The following operator Λ_M is also one of their applications.

Definition 2 *For any integer $M \geq 1$ and any unitary operator $\mathbf{U}$, the operator $\Lambda_M(\mathbf{U})$ is defined by*

$$|j\rangle|y\rangle \ \longmapsto \ \begin{cases} |j\rangle \mathbf{U}^j |y\rangle & (0 \leq j < M) \\ |j\rangle \mathbf{U}^M |y\rangle & (j \geq M). \end{cases}$$

Λ_M is controlled by the first register $|j\rangle$ in this case. $\Lambda_M(\mathbf{U})$ uses $\mathbf{U}$ for M times.

It is also known that quantum transformations can compute all classical functions. Let g be any classically computable function with m input and k output bits. Then, there exists a unitary transformation $\mathbf{U}_g$ corresponding to the computation of g: for any $x \in \{0,1\}^m$ and $y \in \{0,1\}^k$, $\mathbf{U}_g$ maps $|x\rangle|y\rangle$ to $|x\rangle|y \oplus g(x)\rangle$, where $\oplus$ denotes the bitwise exclusive-OR.

2.2 Query Complexity

In this paper, we are interested in the query complexity, which is discussed in the following model. Suppose we want to compute some function $\mathcal{F}$ with an N-bit input and we can access each bit only through a given oracle O. The query complexity is the number of queries to the oracle. A quantum algorithm with T queries is a sequence of unitary transformations: $U_0 \to O_1 \to U_1 \to \ldots \to O_T \to U_T$, where O_i denotes the unitary transformation corresponding to the i-th query to the oracle O, and U_i denotes an arbitrary unitary transformation independent of the oracle. Our natural goal is to find an algorithm to compute $\mathcal{F}$ with sufficiently large probability and with the smallest number of oracle calls.

The most natural quantum oracles are quantum perfect oracles O_f that map $|x\rangle|0^{m-1}\rangle|0\rangle$ to $|x\rangle|0^{m-1}\rangle|f(x)\rangle$ for any $x \in [N]$. Here, $|0^{m-1}\rangle$ is a work register that is always cleared before and after querying oracles. On the other hand, quantum biased oracles, which we deal with in this paper, are defined as follows.

Definition 3 *A quantum oracle of a Boolean function f with bias ε is a unitary transformation O_f^ε or its inverse $O_f^{\varepsilon\dagger}$ such that*

$$O_f^\varepsilon |x\rangle|0^{m-1}\rangle|0\rangle = |x\rangle(\alpha_x|w_x\rangle|f(x)\rangle + \beta_x|w'_x\rangle|\overline{f(x)}\rangle),$$

where $|\alpha_x|^2 = 1/2 + \varepsilon_x \geq 1/2 + \varepsilon$ for any $x \in [N]$. Let also $\varepsilon_{\min} = \min_x \varepsilon_x$.

Note that $0 < \varepsilon \le \varepsilon_{\min} \le \varepsilon_x \le 1/2$ for any x. In practice, ε is usually given in some way and $\varepsilon_{\min}$ or ε_x may be unknown. Unless otherwise stated, we discuss the query complexity with a given biased oracle O_f^ε in the rest of the paper.

We can also consider phase flip oracles instead of the above-defined bit flip oracles. A (perfect) phase flip oracle is defined as a map: $|x\rangle|0^{m-1}\rangle \longmapsto (-1)^{f(x)}|x\rangle|0^{m-1}\rangle$, which is equivalent to the corresponding bit flip oracle O_f in the perfect case, since either oracle can be easily simulated by the other oracle with a pair of Hadmard gates. In a biased case, however, the two oracles cannot always be converted to each other. We need to take care of interference of the work registers, i.e., $|w_x\rangle$ and $|w'_x\rangle$, which are dealt with carefully in our algorithm.

2.3 Amplitude Amplification and Estimation

We briefly introduce a few known quantum algorithms often used in the following sections. In [7], Brassard et al. presented amplitude amplification as follows.

Theorem 1 *Let $\mathcal{A}$ be any quantum algorithm that uses no measurements and $\chi : \mathbb{Z} \to \{0,1\}$ be any Boolean function that distinguishes between success or fail (good or bad). There exists a quantum algorithm that given the initial success probability $p > 0$ of $\mathcal{A}$, finds a good solution with certainty using a number of applications of $\mathcal{A}$ and $\mathcal{A}^{-1}$, which is in $O(\frac{1}{\sqrt{p}})$ in the worst case.*

In the amplitude amplification, a unitary operator $\mathbf{Q} = -\mathcal{A}\mathbf{S}_0\mathcal{A}^{-1}\mathbf{S}_\chi$ is used. Here, $\mathbf{S}_0$ denotes an operator to flip the sign of amplitude of the state $|\mathbf{0}\rangle$, and $\mathbf{S}_\chi$ denotes an operator to flip the signs of amplitudes of all the good states. Applying $\mathbf{Q}$ to the state $\mathcal{A}|\mathbf{0}\rangle$ for j times, we have

$$\mathbf{Q}^j\mathcal{A}|\mathbf{0}\rangle = \frac{1}{\sqrt{p}}\sin((2j+1)\theta_p)\,|\Psi_1\rangle + \frac{1}{\sqrt{1-p}}\cos((2j+1)\theta_p)\,|\Psi_0\rangle,$$

where $|\Psi_1\rangle$ has all the good states, and $\langle\Psi_1|\Psi_1\rangle = p = \sin^2(\theta_p)$ and $|\Psi_1\rangle$ is orthogonal to $|\Psi_0\rangle$. After applying $\mathbf{Q}$ for about $\pi/4\theta_p \in O(1/\sqrt{p})$ times, we can measure a good solution with probability close to 1. Note that we need to know information about the value of p in some way to do so. See [7] for more details.

Brassard et al. also presented amplitude estimation in [7]. We rewrite it in terms of phase estimation as follows.

Theorem 2 *Let $\mathcal{A}, \chi$ and p be as in Theorem 1 and $\theta_p = \sin^{-1}(\sqrt{p})$ such that $0 \le \theta_p \le \pi/2$. There exists a quantum algorithm Est_Phase$(\mathcal{A}, \chi, M)$ that outputs $\tilde{\theta}_p$ such that $|\theta_p - \tilde{\theta}_p| \le \frac{\pi}{M}$, with probability at least $8/\pi^2$. It uses exactly M invocations of $\mathcal{A}$ and χ, respectively. If $\theta_p = 0$ then $\tilde{\theta}_p = 0$ with certainty, and if $\theta_p = \pi/2$ and M is even, then $\tilde{\theta}_p = \pi/2$ with certainty.*

3 Computing OR with ε-Biased Oracles

In this section, we assume that we have information about bias rate of the given biased oracle: a value of ε such that $0 < \varepsilon \le \varepsilon_{\min}$. Under this assumption, in Theorem 3 we show that N-bit OR functions can be computed by using $O(\sqrt{N}/\varepsilon)$ queries to the given oracle O_f^ε. Moreover, when we know $\varepsilon_{\min}$, we can present an optimal algorithm to compute OR with O_f^ε. Before describing the main theorem, we present the following key lemma.

Lemma 1 *There exists a quantum algorithm that simulates a single query to an oracle $O_f^{1/6}$ by using $O(1/\varepsilon)$ queries to O_f^ε if we know ε.*

To prove the lemma, we replace the given oracle O_f^ε with a new oracle $\tilde{O}_f^\varepsilon$ for our convenience. The next lemma describes the oracle $\tilde{O}_f^\varepsilon$ and how to construct it from O_f^ε.

Lemma 2 *There exists a quantum oracle $\tilde{O}_f^\varepsilon$ that consists of one O_f^ε and one ${O_f^\varepsilon}^\dagger$ such that for any $x \in [N]$*

$$\tilde{O}_f^\varepsilon|x, 0^m, 0\rangle = (-1)^{f(x)} 2\varepsilon_x |x, 0^m, 0\rangle + |x, \psi_x\rangle, \tag{1}$$

where $|x, \psi_x\rangle$ is orthogonal to $|x, 0^m, 0\rangle$ and its norm is $\sqrt{1 - 4{\varepsilon_x}^2}$.

Proof. We can show the construction of $\tilde{O}_f^\varepsilon$ in a similar way in Lemma 1 in [13]. □

□

Now, we describe our approach to Lemma 1. The oracle $O_f^{1/6}$ is simulated by the given oracle O_f^ε based on the following idea. According to [13], if the query register $|x\rangle$ is not in a superposition, phase flip oracles can be simulated with sufficiently large probability: by using amplitude estimation through $\tilde{O}_f^\varepsilon$, we can estimate the value of ε_x, then by using the estimated value and applying amplitude amplification to the state in (1), we can obtain the state $(-1)^{f(x)}|x, 0^m, 0\rangle$ with high probability. In Lemma 1, we essentially simulate the phase flip oracle by using the above algorithm in a superposition of $|x\rangle$. Note that we convert the phase flip oracle into the bit flip version in the lemma.

We will present the proof of Lemma 1 after the following lemma, which shows that amplitude estimation can work in quantum parallelism. *Est_Phase* in Theorem 2 is straightforwardly extended to *Par_Est_Phase* in Lemma 3, whose proof can be found in the Appendix.

Lemma 3 *Let $\chi : \mathbb{Z} \to \{0, 1\}$ be any Boolean function, and let $\mathcal{O}$ be any quantum oracle that uses no measurements such that*

$$\mathcal{O}|x\rangle|\mathbf{0}\rangle = |x\rangle\mathcal{O}_x|\mathbf{0}\rangle = |x\rangle|\Psi_x\rangle = |x\rangle(|\Psi_x^1\rangle + |\Psi_x^0\rangle),$$

where a state $|\Psi_x\rangle$ is divided into a good state $|\Psi_x^1\rangle$ and a bad state $|\Psi_x^0\rangle$ by χ. Let $\sin^2(\theta_x) = \langle\Psi_x^1|\Psi_x^1\rangle$ be the success probability of $\mathcal{O}_x|\mathbf{0}\rangle$ where $0 \le \theta_x \le \pi/2$. There exists a quantum algorithm Par_Est_Phase$(\mathcal{O}, \chi, M)$ that changes states as follows:

$$|x\rangle|\mathbf{0}\rangle|\mathbf{0}\rangle \longmapsto |x\rangle \otimes \sum_{j=0}^{M-1} \delta_{x,j}|v_{x,j}\rangle|\tilde{\theta}_{x,j}\rangle,$$

where $\displaystyle\sum_{j:|\theta_x - \tilde{\theta}_{x,j}| \le \frac{\pi}{M}} |\delta_{x,j}|^2 \ge \frac{8}{\pi^2}$ *for any x, and $|v_{x,i}\rangle$ and $|v_{x,j}\rangle$ are mutually orthonormal vectors for any i, j. It uses $\mathcal{O}$ and its inverse for $O(M)$ times.*

Proof. (of Lemma 1)

We will show a quantum algorithm that changes states as follows:

$$|x\rangle|\mathbf{0}\rangle|0\rangle \longmapsto |x\rangle(\alpha_x|w_x\rangle|f(x)\rangle + \beta_x|w'_x\rangle|\overline{f(x)}\rangle),$$

where $|\alpha_x|^2 \ge 2/3$ for any x, using $O(1/\varepsilon)$ queries to O_f^ε. The algorithm performs amplitude amplification following amplitude estimation in a superposition of $|x\rangle$.

At first, we use amplitude estimation in parallel to estimate ε_x or to know how many times the following amplitude amplification procedures should be repeated. Let $\sin\theta = 2\varepsilon$ and $\sin\theta_x = 2\varepsilon_x$ such that $0 < \theta, \theta_x \le \pi/2$. Note that $\Theta(\theta) = \Theta(\varepsilon)$ since $\sin\theta \le \theta \le \frac{\pi}{2}\sin\theta$ when $0 \le \theta \le \pi/2$. Let also $M_1 = \left\lceil \frac{3\pi(\pi+1)}{\theta} \right\rceil$ and χ be a Boolean function that divides a state in (1) into a good state $(-1)^{f(x)} 2\varepsilon_x |0^{m+1}\rangle$ and a bad state $|\psi_x\rangle$. The function χ checks only whether the state is $|0^{m+1}\rangle$ or not; therefore, it is implemented easily. By Lemma 3, $Par\_Est\_Phase(\tilde{O}_f^{\varepsilon}, \chi, M_1)$ maps

$$|x\rangle|\mathbf{0}\rangle|\mathbf{0}\rangle|\mathbf{0}\rangle \longmapsto |x\rangle \otimes \sum_{j=0}^{M-1} \delta_{x,j}|v_{x,j}\rangle|\tilde{\theta}_{x,j}\rangle|\mathbf{0}\rangle,$$

where $\displaystyle\sum_{j:|\theta_x - \tilde{\theta}_{x,j}| \le \frac{\theta}{3(\pi+1)}} |\delta_{x,j}|^2 \ge \frac{8}{\pi^2}$ for any x, and $|v_{x,i}\rangle$ and $|v_{x,j}\rangle$ are mutually orthonormal vectors for any i, j. This state has the good estimations of θ_x in the third register with high probability. The fourth register $|\mathbf{0}\rangle$ remains large enough to perform the following steps.

The remaining steps basically perform amplitude amplification by using the estimated values $\tilde{\theta}_{x,j}$, which can realize a phase flip oracle. Note that in the following steps a pair of Hadmard transformations are used to convert the phase flip oracle into our targeted oracle.

Based on the de-randomization idea as in [13], we calculate $m^*_{x,j} = \left\lceil \frac{1}{2}\left(\frac{\pi}{2\tilde{\theta}_{x,j}} - 1\right)\right\rceil$, $\theta^*_{x,j} = \frac{\pi}{4m^*_{x,j}+2}$, $p^*_{x,j} = \sin^2(\theta^*_{x,j})$ and $\tilde{p}_{x,j} = \sin^2(\tilde{\theta}_{x,j})$ in the superposition, and apply an Hadmard transformation to the last qubit. Thus we have

$$|x\rangle\left(\sum_{j=0}^{M-1} \delta_{x,j}|v_{x,j}\rangle|\tilde{\theta}_{x,j}\rangle|m^*_{x,j}\rangle|\theta^*_{x,j}\rangle|p^*_{x,j}\rangle|\tilde{p}_{x,j}\rangle \otimes |0^{m+1}\rangle|0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right).$$

Next, let $\mathbf{R} : |p^*_{x,j}\rangle|\tilde{p}_{x,j}\rangle|0\rangle \to |p^*_{x,j}\rangle|\tilde{p}_{x,j}\rangle\left(\sqrt{\frac{p^*_{x,j}}{\tilde{p}_{x,j}}}|0\rangle + \sqrt{1 - \frac{p^*_{x,j}}{\tilde{p}_{x,j}}}|1\rangle\right)$ be a rotation and let $\mathbf{O} = \tilde{O}_f^{\varepsilon} \otimes \mathbf{R}$ be a new oracle. We apply $\mathbf{O}$ followed by $\Lambda_{M_2}(\mathbf{Q})$, where $M_2 = \left\lceil \frac{1}{2}\left(\frac{3\pi(\pi+1)}{2(3\pi+2)\theta} + 1\right)\right\rceil$ and $\mathbf{Q} = -\mathbf{O}(\mathbf{I} \otimes \mathbf{S}_0)\mathbf{O}^{-1}(\mathbf{I} \otimes \mathbf{S}_\chi)$; $\mathbf{S}_0$ and $\mathbf{S}_\chi$ are defined appropriately. Λ_{M_2} is controlled by the register $|m^*_{x,j}\rangle$, and $\mathbf{Q}$ is applied to the registers $|x\rangle$ and $|0^{m+1}\rangle|0\rangle$ if the last qubit is $|1\rangle$. Let $\mathbf{O}_x$ denote the unitary operator such that $\mathbf{O}|x\rangle|0^{m+1}\rangle|0\rangle = |x\rangle\mathbf{O}_x|0^{m+1}\rangle|0\rangle$. Then we have the state (From here, we write only the last three registers.)

$$\sum_{j=0}^{M-1} \frac{\delta_{x,j}}{\sqrt{2}}\left(|0^{m+1}\rangle|0\rangle|0\rangle + \mathbf{Q}_x^{m_{x,j}}\mathbf{O}_x\left(|0^{m+1}\rangle|0\rangle\right)|1\rangle\right), \qquad (2)$$

where $\mathbf{Q}_x = -\mathbf{O}_x\mathbf{S}_0\mathbf{O}_x^{-1}\mathbf{S}_\chi$ and $m_{x,j} = \min(m^*_{x,j}, M_2)$ for any x, j. We will show that the phase flip oracle is simulated if the third register $|\tilde{\theta}_{x,j}\rangle$ has the good estimation of θ_x and the last register has $|1\rangle$. Equation (2) can be rewritten as

$$\sum_{j=0}^{M-1} \frac{\delta_{x,j}}{\sqrt{2}}\left(|0^{m+1}, 0\rangle|0\rangle + \left((-1)^{f(x)}\gamma_{x,j}|0^{m+1}, 0\rangle + |\varphi_{x,j}\rangle\right)|1\rangle\right),$$

where $|\varphi_{x,j}\rangle$ is orthogonal to $|0^{m+1}, 0\rangle$ and its norm is $\sqrt{1 - \gamma_{x,j}^2}$. Suppose that the third register has $|\tilde{\theta}_{x,j}\rangle$ such that $|\theta_x - \tilde{\theta}_{x,j}| \le \frac{\theta_x}{3(\pi+1)}$. It can be seen that $m_{x,j} \le M_2$ if $|\theta_x - \tilde{\theta}_{x,j}| \le \frac{\theta_x}{3(\pi+1)}$. Therefore,

$\mathbf{Q}_x$ is applied for $m^*_{x,j}$ times, i.e., the number specified by the fourth register. Like the analysis of Lemma 2 in [13], it is shown that $\gamma_{x,j} \geq \sqrt{1-\frac{1}{9}}$.

Finally, applying an Hadmard transformation to the last qubit again, we have the state

$$\sum_{j=0}^{M-1} \frac{\delta_{x,j}}{2} \Big(\quad (1+(-1)^{f(x)}\gamma_{x,j})|0^{m+2}\rangle|0\rangle$$
$$+ (1-(-1)^{f(x)}\gamma_{x,j})|0^{m+2}\rangle|1\rangle + |\varphi_{x,j}\rangle(|0\rangle - |1\rangle) \Big).$$

If we measure the last qubit, we have $|f(x)\rangle$ with probability

$$\sum_{j=0}^{M-1} \left(\left| \frac{\delta_{x,j}(1+\gamma_{x,j})}{2} \right|^2 + \left| \frac{\delta_{x,j}\sqrt{1-\gamma_{x,j}^2}}{2} \right|^2 \right)$$
$$\geq \frac{1}{2} \sum_{j:|\theta_x - \tilde{\theta}_{x,j}| \leq \frac{\theta}{3(\pi+1)}} |\delta_{x,j}|^2 (1+\gamma_{x,j}) \geq \frac{2}{3}.$$

Thus, the final quantum state can be rewritten as $|x\rangle(\alpha_x|w_x\rangle|f(x)\rangle + \beta_x|w'_x\rangle|\overline{f(x)}\rangle$, where $|\alpha_x|^2 \geq 2/3$ for any x.

The query complexity of this algorithm is the cost of amplitude estimation M_1 and amplitude amplification M_2, thus a total number of queries is $O(\frac{1}{\theta}) = O(\frac{1}{\varepsilon})$. Therefore, we can simulate a single query to $O_f^{1/6}$ using $O(\frac{1}{\varepsilon})$ queries to O_f^{ε}. □

□

Now, we describe the main theorem to compute OR functions with quantum biased oracles.

Theorem 3 *There exists a quantum algorithm to compute N-bit OR with probability at least $2/3$ using $O(\sqrt{N}/\varepsilon)$ queries to a given oracle O_f^{ε} if we know ε. Moreover, if we know $\varepsilon_{\min}$, the algorithm uses $\Theta(\sqrt{N}/\varepsilon_{\min})$ queries.*

Proof. The upper bound $O(\sqrt{N}/\varepsilon)$ is obtained by Lemma 1 and [12]. In [12], we can see an algorithm to compute OR with probability at least 2/3 using $O(\sqrt{N})$ queries to an oracle $O_f^{1/6}$. When an oracle O_f^{ε} and a value of ε are given, we can simulate one query to an oracle $O_f^{1/6}$ using $O(1/\varepsilon)$ queries to O_f^{ε} by Lemma 1. Therefore, we can compute OR using $O(\sqrt{N}/\varepsilon)$ queries to an oracle O_f^{ε}.

The lower bound $\Omega(\sqrt{N}/\varepsilon_{\min})$ is also obtained by Theorem 6 in [13]. The theorem states that for any problem, if the lower bound $\Omega(T)$ can be shown by Ambainis' method in the noiseless case, then the lower bound $\Omega(T/\varepsilon_{\min})$ can also be shown in the noisy case. For computing N-bit OR functions, $\Omega(\sqrt{N})$ can be shown by Ambainis' method; therefore, we can derive $\Omega(\sqrt{N}/\varepsilon_{\min})$ in the quantum biased setting. □

□

4 Estimating Unknown ε

In Sect.3, we described algorithms by using a given oracle O_f^{ε} when we know ε. In this section, we assume that there is no prior knowledge of ε.

Our overall approach is to estimate ε with appropriate accuracy (in precise $\varepsilon_{\min}$) in advance, which then can be used in the simulating algorithm in Lemma 1. In the following, we first describe an overview of our strategy to estimate $\varepsilon_{\min}$ rather informally, followed by rigorous and detailed descriptions.

First, let us consider estimating ε_x in the same way as Lemma 1 in quantum parallelism. Then, let M^* denote the number of required oracle calls to achieve a *good* estimation of ε_x for any x. (Here, *good* means accurate enough to perform effective amplitude amplification in Lemma 1.) Note that $M^* \in \Omega(1/\varepsilon_{\min})$, and if we know the value of ε, we can set $\Theta(1/\varepsilon)$ as M^*. However, now ε is unknown, we estimate M^* as follows. First we will construct an algorithm, $\mathcal{A}_{\text{enough}}$, which receives an input M and decides whether M is the number of oracle calls to obtain a good estimation of ε_x. More precisely, $\mathcal{A}_{\text{enough}}$ uses $O(M)$ queries and returns 0 if the input M is large enough to estimate ε_x, otherwise it returns 1 with a more than constant probability, say, 9/10. Then, by using $\mathcal{A}_{\text{enough}}$ in a superposition of $|x\rangle$ as in Lemma 4, we can obtain the state $\sum_x |x\rangle \otimes (\alpha_x|u_x\rangle|1\rangle + \beta_x|u'_x\rangle|0\rangle)$. When M is small, the condition $\exists x; |\alpha_x|^2 \geq 9/10$ holds, which means there exists x such that the estimation of ε_x may be bad. On the other hand, when M is sufficiently large, the condition $\forall x; |\alpha_x|^2 \leq 1/10$ holds, which means the estimation is good for any x. Our remaining essential task, then, is to know an input value of M at the verge of the above two cases. Note that the value is $\Theta(1/\varepsilon_{\min})$, which can be used as M^*.

Next, we consider an algorithm, A_{check}, which can distinguish the above two cases with $O(T)$ oracle queries with a constant probability. Then, M^* can be estimated by $O(TM^* \log\log M^*)$ queries by the following search technique and majority voting: We can find M^* by trying $\mathcal{A}_{\text{check}}$ along with exponentially increasing the input value M until $\mathcal{A}_{\text{check}}$ succeeds. Note that a $\log\log M^*$ factor is needed to boost the success probability of $\mathcal{A}_{\text{check}}$ to close to 1. It should be noted that we cannot use robust quantum search algorithm [12] as $\mathcal{A}_{\text{check}}$, since there may exist x such that $|\alpha_x|^2 \approx 1/2$, which cannot be dealt with by their algorithm. Instead, in Lemma 5, we will describe the algorithm $\mathcal{A}_{\text{check}}$, which can distinguish the above two cases by using amplitude estimation querying for $O(\sqrt{N}\log N)$ times. Then, the whole algorithm requires $O(TM^* \log\log M^*) = O\left(\frac{\sqrt{N}\log N}{\varepsilon_{\min}} \log\log \frac{1}{\varepsilon_{\min}}\right)$ queries. In Lemma 4, we present an algorithm *Par_Est_Zero* that acts as $\mathcal{A}_{\text{enough}}$ in a superposition of $|x\rangle$, and in Lemma 5, we describe the algorithm *Chk_Amp_Dn* as $\mathcal{A}_{\text{check}}$. Finally, the whole algorithm to estimate M^* is presented in Theorem 4.

Lemma 4 *Let $\mathcal{O}$ be any quantum algorithm that uses no measurements such that $\mathcal{O}|x\rangle|\mathbf{0}\rangle = |x\rangle|\Psi_x\rangle = |x\rangle(|\Psi_x^1\rangle + |\Psi_x^0\rangle)$. Let $\chi : \mathbb{Z} \to \{0,1\}$ be a Boolean function that divides a state $|\Psi_x\rangle$ into a good state $|\Psi_x^1\rangle$ and a bad state $|\Psi_x^0\rangle$ such that $\sin^2(\theta_x) = \langle\Psi_x^1|\Psi_x^1\rangle$ for any x ($0 < \theta_x \leq \pi/2$). There exists a quantum algorithm $Par\_Est\_Zero(\mathcal{O}, \chi, M)$ that changes states as follows:*

$$|x\rangle|\mathbf{0}\rangle|0\rangle \to |x\rangle \otimes (\alpha_x|u_x\rangle|1\rangle + \beta_x|u'_x\rangle|0\rangle)\,,$$

where $|\alpha_x|^2 = \dfrac{\sin^2(M\theta_x)}{M^2\sin^2(\theta_x)}$ *for any x. It uses $\mathcal{O}$ and its inverse for $O(M)$ times.*

Proof. The algorithm $Par\_Est\_Zero(\mathcal{O}, \chi, M)$ acts as $Par\_Est\_Phase(\mathcal{O}, \chi, M)$ from Step 1 to Step 5, and applies a unitary transformation corresponding to the following function $g'(x)$ instead of $g_M(x)$ at Step 6,

$$g'(x) = \begin{cases} 1 & (x = 0) \\ 0 & (otherwise). \end{cases}$$

Then, like (3) we have the state

$$
\begin{aligned}
|x\rangle \quad &\otimes \quad \frac{-\imath}{\sqrt{2}}\Bigg(e^{\imath\theta_x}|\Psi_x^+\rangle\Bigg(\alpha_{x,0}^+|0\rangle|1\rangle + \sum_{j=1}^{M-1}\alpha_{x,j}^+|j\rangle|0\rangle\Bigg) \\
&- \quad e^{-\imath\theta_x}|\Psi_x^-\rangle\Bigg(\alpha_{x,0}^-|0\rangle|1\rangle + \sum_{j=1}^{M-1}\alpha_{x,j}^-|j\rangle|0\rangle\Bigg)\Bigg),
\end{aligned}
$$

where $|\alpha_{x,j}^{\pm}|^2 = \frac{\sin^2(M\Delta_{x,j}^{\pm}\pi)}{M^2\sin^2(\Delta_{x,j}^{\pm}\pi)}$ such that $\Delta_{x,j}^+ = d(\frac{j}{M}, \frac{\theta_x}{\pi})$ and $\Delta_{x,j}^- = d(\frac{j}{M}, 1-\frac{\theta_x}{\pi})$ for any x, j. (Precisely speaking, $|\alpha_{x,j}^{\pm}|^2 = 1$ when $\Delta_{x,j}^{\pm} = 0$. However, $\Delta_{x,0}^{\pm} \neq 0$ since $\theta_x \neq 0$ in this case.) Note that $|\Psi_x^+\rangle$ and $|\Psi_x^-\rangle$ are mutually orthogonal and $\langle\Psi_x^{\pm}|\Psi_x^{\pm}\rangle = 1$. Therefore, for any x the last qubit has $|1\rangle$ with probability

$$
\frac{|\alpha_{x,0}^+|^2}{2} + \frac{|\alpha_{x,0}^-|^2}{2} = \frac{\sin^2(M\theta_x)}{M^2\sin^2(\theta_x)}.
$$

Par_Est_Zero$(\mathcal{O}, \chi, M)$ requires $O(M)$ queries to $\mathcal{O}$. They are used when the algorithm is working as *Par_Est_Phase*$(\mathcal{O}, \chi, M)$. □

□

Lemma 5 *Let $\mathcal{O}$ be any quantum oracle such that $\mathcal{O}|x\rangle|\mathbf{0}\rangle|0\rangle = |x\rangle(\alpha_x|w_x\rangle|1\rangle + \beta_x|u_x\rangle|0\rangle)$. There exists a quantum algorithm Chk_Amp_Dn$(\mathcal{O})$ that outputs $b \in \{0,1\}$ such that*

$$
b = \begin{cases} 1 & if \;\; \exists x; |\alpha_x|^2 \geq \frac{9}{10} \\ 0 & if \;\; \forall x; |\alpha_x|^2 \leq \frac{1}{10} \\ don't\ care & otherwise, \end{cases}
$$

with probability at least $8/\pi^2$ using $O(\sqrt{N}\log N)$ queries to $\mathcal{O}$.

Proof. Using $O(\log N)$ applications of $\mathcal{O}$ and majority voting, we have a new oracle $\mathcal{O}'$ such that

$$
\mathcal{O}'|x\rangle|\mathbf{0}\rangle|0\rangle = |x\rangle(\alpha_x'|w_x'\rangle|1\rangle + \beta_x'|u_x'\rangle|0\rangle),
$$

where $|\alpha_x'|^2 \geq 1 - \frac{1}{16N}$ if $|\alpha_x|^2 \geq \frac{9}{10}$, and $|\alpha_x'|^2 \leq \frac{1}{16N}$ if $|\alpha_x|^2 \leq \frac{1}{10}$. Note that work bits $|w_x'\rangle$ and $|u_x'\rangle$ are likely larger than $|w_x\rangle$ and $|u_x\rangle$.

Now, let $\mathcal{A}$ be a quantum algorithm that makes the uniform superposition $\frac{1}{\sqrt{N}}\sum_x |x\rangle|\mathbf{0}\rangle|0\rangle$ by the Fourier transform $\mathbf{F}_N$ and applies the oracle $\mathcal{O}'$. We consider (success) probability p that the last qubit in the final state $\mathcal{A}|\mathbf{0}\rangle$ has $|1\rangle$. If the given oracle $\mathcal{O}$ satisfies $\exists x; |\alpha_x|^2 \geq \frac{9}{10}$ (we call Case 1), the probability p is at least $\frac{1}{N} \times (1 - \frac{1}{16N}) \geq \frac{15}{16N}$. On the other hand, if $\mathcal{O}$ satisfies $\forall x; |\alpha_x|^2 \leq \frac{1}{10}$ (we call Case 2), then the probability $p \leq N \times \frac{1}{N} \times \frac{1}{16N} = \frac{1}{16N}$. We can distinguish the two cases by amplitude estimation as follows.

Let $\tilde{\theta}_p$ denote the output of the amplitude estimation *Est_Phase*$(\mathcal{A}, \chi, \lceil 11\sqrt{N}\rceil)$. The whole algorithm *Chk_Amp_Dn*$(\mathcal{O})$ performs *Est_Phase*$(\mathcal{A}, \chi, \lceil 11\sqrt{N}\rceil)$ and outputs whether $\tilde{\theta}_p$ is greater than $0.68/\sqrt{N}$ or not. We will show that it is possible to distinguish the above two cases by the value of $\tilde{\theta}_p$. Let $\theta_p = \sin^{-1}(\sqrt{p})$ such that $0 \leq \theta_p \leq \pi/2$. Note that $x \leq \sin^{-1}(x) \leq \pi x/2$ if $0 \leq x \leq 1$. Theorem 2 says that in Case 1, the *Est_Phase* outputs $\tilde{\theta}_p$ such that

$$
\tilde{\theta}_p \geq \theta_p - \frac{\pi}{11\sqrt{N}} \geq \sqrt{\frac{15}{16N}} - \frac{\pi}{11\sqrt{N}} > \frac{0.68}{\sqrt{N}},
$$

with probability at least $8/\pi^2$. Similarly in Case 2, the inequality $\tilde{\theta}_p < \frac{0.68}{\sqrt{N}}$ is obtained.

$Chk\_Amp\_Dn(\mathcal{O})$ uses $\mathcal{O}$ for $O(\sqrt{N}\log N)$ times since $Chk\_Amp\_Dn(\mathcal{O})$ calls the algorithm $\mathcal{A}$ for $\lceil 11\sqrt{N}\rceil$ times and $\mathcal{A}$ uses $O(\log N)$ queries to the given oracle $\mathcal{O}$. □

□

Theorem 4 *Given a quantum biased oracle O_f^ε, there exists a quantum algorithm $Est\_Eps\_Min(O_f^\varepsilon)$ that outputs $\tilde{\varepsilon}_{\min}$ such that $\varepsilon_{\min}/5\pi^2 \leq \tilde{\varepsilon}_{\min} \leq \varepsilon_{\min}$ with probability at least 2/3. The query complexity of the algorithm is expected to be $O\left(\frac{\sqrt{N}\log N}{\varepsilon_{\min}}\log\log\frac{1}{\varepsilon_{\min}}\right)$.*

Proof. Let $\sin(\theta_x) = 2\varepsilon_x$ and $\sin(\theta_{\min}) = 2\varepsilon_{\min}$ such that $0 < \theta_x, \theta_{\min} \leq \frac{\pi}{2}$. Let χ also be a Boolean function that divides the state in (1) into a good state $(-1)^{f(x)}2\varepsilon_x|0^{m+1}\rangle$ and a bad state $|\psi_x\rangle$. Thus $Par\_Est\_Zero(\tilde{O}_f^\varepsilon, \chi, M)$ in Lemma 4 makes the state $|x\rangle \otimes (\alpha_x|u_x\rangle|1\rangle + \beta_x|u'_x\rangle|0\rangle)$ such that $|\alpha_x|^2 = \frac{\sin^2(M\theta_x)}{M^2\sin^2(\theta_x)}$. As stated below, if $M \in o(1/\theta_x)$, then $|\alpha_x|^2 \geq 9/10$. We can use Chk_Amp_Dn to check whether there exists x such that $|\alpha_x|^2 \geq 9/10$. Based on these facts, we present the whole algorithm $Est\_Eps\_Min(O_f^\varepsilon)$.

Algorithm($Est\_Eps\_Min(O_f^\varepsilon)$ **)**

1. Start with $\ell = 0$.
2. Increase ℓ by 1.
3. Run $Chk\_Amp\_Dn(Par\_Est\_Zero(\tilde{O}_f^\varepsilon, \chi, 2^\ell))$ for $O(\log\ell)$ times and use majority voting. If "1" is output as the result of the majority voting, then return to Step 2.
4. Output $\tilde{\varepsilon}_{\min} = \frac{1}{2}\sin\left(\frac{1}{5\cdot 2^\ell}\right)$.

Now, we will show that the algorithm almost keeps running until $\ell > \left\lfloor \log_2\frac{1}{5\theta_{\min}}\right\rfloor$. We assume $\ell \leq \left\lfloor \log_2\frac{1}{5\theta_{\min}}\right\rfloor$. Under this assumption, a proposition $\exists x; |\alpha_x|^2 \geq \frac{9}{10}$ holds since the equation $\varepsilon_{\min} = \min_x \varepsilon_x$ guarantees that there exists some x such that $\theta_{\min} = \theta_x$ and $|\alpha_x|^2 = \frac{\sin^2(2^\ell\theta_x)}{2^{2\ell}\sin^2(\theta_x)} \geq \cos^2(\frac{1}{5}) > \frac{9}{10}$ when $2^\ell \leq \frac{1}{5\theta_x}$. Therefore, a single Chk_Amp_Dn run returns "1" with probability at least $8/\pi^2$. By $O(\log\ell)$ repetitions and majority voting, the probability that we obtain "1" increases to at least $1 - \frac{1}{5\ell^2}$. Consequently, the overall probability that we return from Step 3 to Step 2 for any ℓ such that $\ell \leq \left\lfloor \log_2\frac{1}{5\theta_{\min}}\right\rfloor$ is at least $\prod_{\ell=1}^{\left\lfloor \log_2\frac{1}{5\theta_{\min}}\right\rfloor}\left(1 - \frac{1}{5\ell^2}\right) > \frac{2}{3}$. This inequality can be obtained by considering an infinite product expansion of $\sin(x)$, i.e., $\sin(x) = x\prod_{n=1}^{\infty}\left(1 - \frac{x^2}{n^2\pi^2}\right)$ at $x = \pi/\sqrt{5}$. Thus the algorithm keeps running until $\ell > \left\lfloor \log_2\frac{1}{5\theta_{\min}}\right\rfloor$, i.e., outputs $\tilde{\varepsilon}_{\min}$ such that $\tilde{\varepsilon}_{\min} = \frac{1}{2}\sin\left(\frac{1}{5\cdot 2^\ell}\right) \leq \frac{1}{2}\sin(\theta_{\min}) = \varepsilon_{\min}$, with probability at least 2/3.

We can also show that the algorithm almost stops in $\ell < \left\lceil \log_2\frac{2\pi}{\theta_{\min}}\right\rceil$. Since $\frac{\sin^2(M\theta)}{M^2\sin^2(\theta)} \leq \frac{\pi^2}{(2M\theta)^2}$ when $0 \leq \theta \leq \frac{\pi}{2}$, $|\alpha_x|^2 = \frac{\sin^2(2^\ell\theta_x)}{2^{2\ell}\sin^2(\theta_x)} \leq \frac{1}{16}$ for any x if $2^\ell \geq \frac{2\pi}{\theta_{\min}}$. Therefore, in Step 3, "0" is returned with probability at least $8/\pi^2$ when $\ell \geq \left\lceil \log_2\frac{2\pi}{\theta_{\min}}\right\rceil$. The algorithm, thus, outputs $\tilde{\varepsilon}_{\min} = \frac{1}{2}\sin\left(\frac{1}{5\cdot 2^\ell}\right) \geq \frac{1}{2}\sin(\frac{\theta_{\min}}{10\pi}) \geq \frac{\varepsilon_{\min}}{5\pi^2}$ with probability at least $8/\pi^2$.

Let $\tilde{\ell}$ satisfy $\left\lfloor \log_2 \frac{1}{5\theta_{\min}} \right\rfloor < \tilde{\ell} < \left\lceil \log_2 \frac{2\pi}{\theta_{\min}} \right\rceil$. If the algorithm runs until $\ell = \tilde{\ell}$, its query complexity is

$$\sum_{\ell=1}^{\tilde{\ell}} O(2^{\ell}\sqrt{N}\log N \log \ell) = O(2^{\tilde{\ell}}\sqrt{N}\log N \log \tilde{\ell}) = O\left(\frac{\sqrt{N}\log N}{\varepsilon_{\min}} \log\log \frac{1}{\varepsilon_{\min}}\right),$$

since $2^{\tilde{\ell}} \in \Theta\left(\frac{1}{\theta_{\min}}\right) = \Theta\left(\frac{1}{\varepsilon_{\min}}\right)$. □

□

Remark. As mentioned above, we have some way to deal with quantum biased oracles even if we have no knowledge about the given oracle's bias rate. On the other hand, in the classical biased setting, there seems to be no way if the value of ε is unknown: Suppose that classical biased oracles return a correct value with probability at least $1/2 + \varepsilon$ for each query. It is known that by using $O(1/\varepsilon^2)$ queries and majority voting, the probability that oracles answer queries correctly increases to 2/3. However, this algorithm works effectively when we know ε. In other words, unless we know ε, it is likely impossible to determine an appropriate number of majority voting to achieve at least a constant success probability of the whole algorithm.

5 Conclusion

In this paper, we have shown that $O(\sqrt{N}/\varepsilon)$ queries are enough to compute N-bit OR with an ε-biased oracle. This matches the known lower bound while affirmatively answering the conjecture raised by the paper [13]. The result in this paper implies other matching bounds such as computing parity with $\Theta(N/\varepsilon)$ queries. We also show a quantum algorithm that estimates unknown value of ε with an ε-biased oracle. Then, by using the estimated value, we can construct a robust algorithm even when ε is unknown. This contrasts with the corresponding classical case where no good estimation method seems to exist.

Until now, unfortunately, we have had essentially only one quantum algorithm, i.e., the robust quantum search algorithm [12], to cope with imperfect oracles. (Note that other algorithms, including our own algorithm in Theorem 3, are all based on the robust quantum search algorithm [12].) Thus, it should be interesting to seek another *essentially different* quantum algorithm with imperfect oracles. If we find a new quantum algorithm that uses $O(T)$ queries to imperfect oracles with constant probability, then we can have a quantum algorithm that uses $O(T/\varepsilon)$ queries to imperfect oracles with an ε-biased oracle based on our method. This is different from the classical case where we need an overhead factor of $O(1/\varepsilon^2)$ by majority voting.

References

[1] S. Aaronson and A. Ambainis. Quantum search of spatial regions. In *Proceedings of 35th ACM Symposium on Theory of Computing*, pages 200–209, 2003.

[2] M. Adcock and R. Cleve. A quantum Goldreich-Levin Theorem with cryptographic applications. In *STACS*, pages 323–334, 2002.

[3] A. Ambainis. Quantum lower bounds by quantum arguments. *J. Comput. Syst. Sci.*, 64(4):750–767, 2002.

[4] A. Ambainis. Quantum walk algorithm for element distinctness. In *Proc. 45th Annual IEEE Symposium on Foudations of Computer Science*, pages 22–31, 2004.

[5] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R de Wolf. Quantum lower bounds by polynomials. In *Proc. 39th Annual IEEE Symposium on Foudations of Computer Science*, pages 352–361, 1998.

[6] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Proc. of the Workshop on Physics of Computation: PhysComp'96*, 1996. LANL preprint, http://xxx.lanl.gov/archive/quant-ph/9605034.

[7] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. In *Quantum Computation & Information*, volume 305 of *AMS Contemporary Mathematics Series Millenium Volume*, pages 53–74, 2002.

[8] H. Buhrman, C. Durr, M. Heiligman, P. Hoyer, F. Magniez, M. Santha, and R. Wolf. Quantum algorithm for element distinctness. In *Proceedings of 16th IEEE Conference on Computational Complexity*, pages 131–137, 2001.

[9] H. Buhrman, I. Newman, H. Röhrig, and R. de Wolf. Robust polynomials and quantum algorithms. In *STACS*, pages 593–604, 2005.

[10] U. Feige, P. Raghavan, D. Peleg, and E. Upfal. Computing with Noisy Information. *SIAM J. Comput.*, 23(5):1001–1018, 1994.

[11] L. K. Grover. A fast quantum mechanical algorithm for database search. In *STOC*, pages 212–219, 1996.

[12] P. Høyer, M. Mosca, and R. de Wolf. Quantum search on bounded-error inputs. In *ICALP*, pages 291–299, 2003.

[13] K. Iwama, R. Raymond, and S. Yamashita. General bounds for quantum biased oracles. *IPSJ Journal*, 46(10):1234–1243, 2005.

[14] Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. In *Proc. 43th Annual IEEE Symposium on Foudations of Computer Science*, pages 513–519, 2002.

[15] P. W. Shor. An algorithm for quantum computation: discrete log and factoring. In *Proc. 35th Annual IEEE Symposium on Foudations of Computer Science*, pages 124–134, 1994.

Appendix

Here, we describe the algorithm and the proof of Lemma 3 after providing a few definitions. The algorithm $Par\_Est\_Phase(\mathcal{O}, \chi, M)$ is based on the amplitude estimation algorithm in [7]. We refer interested readers to [7].

Definition 4 *For any two real numbers $\omega_0, \omega_1 \in \mathbb{R}$,*

$$d(\omega_0, \omega_1) = \min_{z \in \mathbb{Z}} \{|z + \omega_1 - \omega_0|\}.$$

Thus $2\pi d(\omega_0, \omega_1)$ is the length of the shortest arc on the unit circle going from $e^{2\pi\imath\omega_0}$ to $e^{2\pi\imath\omega_1}$. Note that $0 \le d(\omega_0, \omega_1) \le \frac{1}{2}$ for any ω_0, ω_1.

Definition 5 *For any integer $M \ge 1$, let $g_M(x)$ be a function defined by*

$$g_M(x) = \begin{cases} \frac{\pi x}{M} & (0 \le x \le \frac{M}{2}) \\ \pi - \frac{\pi x}{M} & (\frac{M}{2} \le x < M). \end{cases}$$

Algorithm(*Par_Est_Phase*$(\mathcal{O}, \chi, M)$ **)**

1. Start with the state $|x\rangle|\mathbf{0}\rangle|\mathbf{0}\rangle|\mathbf{0}\rangle$.
2. Apply $\mathcal{O}$ to the first and the second registers.
3. Apply $\mathbf{F}_M$ to the third register.
4. Apply $\Lambda_M(\mathbf{Q})$ controlled by the third register, where $\mathbf{Q} = -\mathcal{O}(\mathbf{I} \otimes \mathbf{S}_0)\mathcal{O}^{-1}(\mathbf{I} \otimes \mathbf{S}_\chi)$. $\mathbf{Q}$ is applied to the first and the second registers.
5. Apply $\mathbf{F}_M^{-1}$ to the third register.
6. Apply the unitary transformation $\mathbf{U}_{g_M}$ to the third and the fourth registers, where $\mathbf{U}_{g_M}$ maps $|x\rangle|\mathbf{0}\rangle$ to $|x\rangle|g_M(x)\rangle$.

Proof. (of Lemma 3)

When $\theta_x = 0, \frac{\pi}{2}$, the analysis can be performed almost like the following; therefore, we assume $0 < \theta_x < \frac{\pi}{2}$ for any x. Focusing on the subspace where the first register has a basis state $|x\rangle$, the transformation $\mathbf{Q}_x^j \mathcal{O}_x$ is applied to the second register, where $\mathbf{Q}_x = -\mathcal{O}_x \mathbf{S}_0 \mathcal{O}_x^{-1} \mathbf{S}_\chi$ and j is the number designated by the third register. Since this situation is the same as in Theorem 12 in [7], the analysis can be done similarly. Let

$$|\Psi_x^\pm\rangle = \frac{1}{\sqrt{2}}\left(\frac{1}{\sin\theta_x}|\Psi_x^1\rangle \pm \frac{\imath}{\cos\theta_x}|\Psi_x^0\rangle\right).$$

Note that $|\Psi_x^+\rangle$ and $|\Psi_x^-\rangle$ are orthonormal eigenvectors of $\mathbf{Q}_x$. After Step 6, we can obtain the state

$$\begin{aligned} |x\rangle \quad \otimes \quad & \frac{-\imath}{\sqrt{2}}\Bigg(e^{\imath\theta_x}|\Psi_x^+\rangle\left(\sum_{j=0}^{M-1}\alpha_{x,j}^+|j\rangle|g_M(j)\rangle\right) \\ & - \quad e^{-\imath\theta_x}|\Psi_x^-\rangle\left(\sum_{j=0}^{M-1}\alpha_{x,j}^-|j\rangle|g_M(j)\rangle\right)\Bigg), \end{aligned} \tag{3}$$

where $|\alpha_{x,j}^\pm|^2 = \frac{\sin^2(M\Delta_{x,j}^\pm\pi)}{M^2\sin^2(\Delta_{x,j}^\pm\pi)}$ such that $\Delta_{x,j}^+ = d(\frac{j}{M}, \frac{\theta_x}{\pi})$ and $\Delta_{x,j}^- = d(\frac{j}{M}, 1-\frac{\theta_x}{\pi})$ for any x, j. (Precisely speaking, $|\alpha_{x,j}^\pm|^2 = 1$ when $\Delta_{x,j}^\pm = 0$. This condition means that $\frac{M\theta_x}{\pi}$ or $M - \frac{M\theta_x}{\pi}$ is an integer.) This follows Theorem 11 in [7].

We will show that the last register has the good estimations of θ_x with high probability in the final state. Now, let $j_1^+ = \lfloor\frac{M\theta_x}{\pi}\rfloor$ and $j_2^+ = \lceil\frac{M\theta_x}{\pi}\rceil$. $0 < \theta_x < \frac{\pi}{2}$ means $0 \le j_i^+ \le \frac{M}{2}$, thus $g_M(j_i^+) = \frac{j_i^+\pi}{M}$

holds. We can prove $|\alpha^+_{x,j_1^+}|^2 + |\alpha^+_{x,j_2^+}|^2 \geq \frac{8}{\pi^2}$ and $|\theta_x - g_M(j_i^+)| \leq \frac{\pi}{M}$ like Theorem 11 in [7]. Similarly, let $j_1^- = M - \lfloor \frac{M\theta_x}{\pi} \rfloor$ and $j_2^- = M - \lceil \frac{M\theta_x}{\pi} \rceil$. $0 < \theta_x < \frac{\pi}{2}$ means $\frac{M}{2} \leq j_i^- \leq M$, thus $g_M(j_i^-) = \pi - \frac{j_i^- \pi}{M}$ holds. (This holds when $\frac{M}{2} \leq j_i^- < M$. When $j_i^- = M$, we consider $j_i^- = 0$, then the following sentences will hold.) We can also prove $|\alpha^-_{x,j_1^-}|^2 + |\alpha^-_{x,j_2^-}|^2 \geq \frac{8}{\pi^2}$ and $|\theta_x - g_M(j_i^-)| \leq \frac{\pi}{M}$. Thus the probability that the last register $|g_M(j)\rangle$ has an estimated value $\tilde{\theta}_x$ such that $|\theta_x - \tilde{\theta}_x| \leq \frac{\pi}{M}$ is

$$\begin{aligned} & \sum_{j:|\theta_x - g_M(j)| \leq \frac{\pi}{M}} \frac{|\alpha^+_{x,j}|^2}{2} + \sum_{j:|\theta_x - g_M(j)| \leq \frac{\pi}{M}} \frac{|\alpha^-_{x,j}|^2}{2} \\ \geq & \sum_{j \in \{j_1^+, j_2^+\}} \frac{|\alpha^+_{x,j}|^2}{2} + \sum_{j \in \{j_1^-, j_2^-\}} \frac{|\alpha^-_{x,j}|^2}{2} \geq \frac{8}{\pi^2}. \end{aligned}$$

Therefore, the well-estimated values of θ_x lie in the last register with probability at least $8/\pi^2$. □